

WE CLAIM:

1. A method of calculating a target processor performance level of a processor from a utilisation history of said processor in performance of a plurality of processing tasks, said method comprising the steps of:

5 calculating a task work value indicating processor utilisation in performing a given processing task within a predetermined task time-interval; and
calculating said target processor performance level in dependence upon said task work value.

10 2. A method as claimed in claim 1, comprising calculating a plurality of task work values corresponding to a respective plurality of previous executions of said given processing task and combining said plurality of task work values to calculate said target processor performance level for a future execution of said given processing task.

15 3. A method as claimed in claim 2, wherein said predetermined task time-interval is independently set for each of said plurality of processing tasks.

4. A method as claimed in claim 3, wherein said predetermined task time-interval
20 is independently set for each execution of said given processing task.

5. A method as claimed in claim 4, wherein said predetermined task time-interval is a time period extending from the start of a first scheduling of said given processing task to the start of a subsequent scheduling of said given processing task, said
25 predetermined task time-interval being associated with said first scheduling.

6. A method as claimed in claim 2, wherein said plurality of task work values corresponding to previous executions of said given processing task are combined to calculate an exponentially decaying average work done value for said given
30 processing task.

7. A method as claimed in claim 1, comprising detecting an idle time duration value within said predetermined task time-interval and calculating a task execution

deadline for said given processing task in dependence upon said task work value and said idle time duration.

8. A method as claimed in claim 7, wherein said task execution deadline is
5 calculated for each of a plurality of previous executions of said given processing task and the plurality of task execution deadlines are combined to calculate an exponentially decaying average task execution deadline value.

9. A method as claimed in claim 7, wherein said target processor performance
10 level for said given processing task is calculated in dependence upon said exponentially decaying average work done value and said exponentially decaying average task execution deadline value corresponding to said given processing task.

10. A method as claimed in claim 1, further comprising the steps of:
15 detecting at least one suspended execution period during processing of said given processing task, said at least one suspended execution period representing a time period during which processing is switched from said given processing task to a further, different processing task prior to completion of said first task; and
calculating said task work value for said given processing task such that it
20 includes processor utilisation during said at least one suspended execution period.

11. A method as claimed in claim 10, comprising setting an upper threshold for said predetermined task time-interval such that if said given processing task continues to execute without detection of said suspended execution period for a duration greater
25 than or equal to said upper threshold, said target processor performance level for said task is automatically recalculated.

12. A method according to claim 1, wherein a flag value is stored for each task, said flag value indicating if the corresponding task has started to execute but has not
30 yet completed execution.

13. A method as claimed in claim 2, wherein each task work value for a respective previous execution of said given processing task is normalised by a corresponding

predetermined task time-interval when combining said task work values to calculate said target processor performance level for said future execution of said task.

14. A computer program product bearing a computer program for controlling a computer to calculate a target processor performance level of a processor from a utilisation history of said processor in performance of a plurality of processing tasks, said computer program comprising:
- task work value calculating code operable to calculate a task work value indicating processor utilisation in performing a given processing task within a predetermined task time-interval; and
- target processor performance calculating code operable to calculate said target processor performance level in dependence upon said task work value.

15. A computer program product as claimed in claim 14, wherein said task work calculating code calculates a plurality of task work values corresponding to a respective plurality of previous executions of said given processing task and combining said plurality of task work values to calculate said target processor performance level for a future execution of said given processing task.

16. A computer program product as claimed in claim 15, wherein said predetermined task time-interval is independently set for each of said plurality of processing tasks.

17. A computer program product as claimed in claim 16, wherein said predetermined task time-interval is independently set for each execution of said given processing task.

18. A computer program product as claimed in claim 17, wherein said predetermined task time-interval is a time period extending from the start of a first scheduling of said given processing task to the start of a subsequent scheduling of said given processing task, said predetermined task time-interval being associated with said first scheduling.

19. A computer program product as claimed in claim 15, wherein said plurality of task work values corresponding to previous executions of said given processing task are combined to calculate an exponentially decaying average work done value for said given processing task.

5

20. A computer program product as claimed in claim 14, comprising detecting code operable to detecting an idle time duration value within said predetermined task time-interval and calculating a task execution deadline for said given processing task in dependence upon said task work value and said idle time duration.

10

21. A computer program product as claimed in claim 20, wherein said task execution deadline is calculated for each of a plurality of previous executions of said given processing task and the plurality of task execution deadlines are combined to calculate an exponentially decaying average task execution deadline value.

15

22. A computer program product as claimed in claim 20, wherein said target processor performance level for said given processing task is calculated in dependence upon said exponentially decaying average work done value and said exponentially decaying average task execution deadline value corresponding to said given processing task.

20

23. A computer program product as claimed in claim 14, further comprising:
suspended execution period detecting code operable to detect at least one suspended execution period during processing of said given processing task, said at least one suspended execution period representing a time period during which processing is switched from said given processing task to a further, different processing task prior to completion of said first task; and

25

wherein said task work value calculating code is operable to calculate said task work value for said given processing task such that it includes processor utilisation during said at least one suspended execution period.

30

24. A computer program product as claimed in claim 23, wherein an upper threshold for said predetermined task time-interval is set such that if said given

processing task continues to execute without detection of said suspended execution period for a duration greater than or equal to said upper threshold, said target processor performance level for said task is automatically recalculated.

5 25. A computer program product according to claim 14, wherein a flag value is stored for each task, said flag value indicating if the corresponding task has started to execute but has not yet completed execution.

10 26. A computer program product as claimed in claim 15, wherein each task work value for a respective previous execution of said given processing task is normalised by a corresponding predetermined task time-interval when combining said task work values to calculate said target processor performance level for said future execution of said task.

15 27. Apparatus for calculating a target processor performance level of a processor from a utilisation history of said processor in performance of a plurality of processing tasks, said apparatus comprising:

task work value calculating logic operable to calculate a task work value indicating processor utilisation in performing a given processing task within a
20 predetermined task time-interval; and
target processor performance calculating logic operable to calculate said target processor performance level in dependence upon said task work value.

25 28. Apparatus as claimed in claim 27, wherein said task work calculating logic calculates a plurality of task work values corresponding to a respective plurality of previous executions of said given processing task and combining said plurality of task work values to calculate said target processor performance level for a future execution of said given processing task.

30 29. Apparatus as claimed in claim 28, wherein said predetermined task time-interval is independently set for each of said plurality of processing tasks.

30. Apparatus as claimed in claim 29, wherein said predetermined task time-interval is independently set for each execution of said given processing task.

31. Apparatus as claimed in claim 30, wherein said predetermined task time-interval is a time period extending from the start of a first scheduling of said given processing task to the start of a subsequent scheduling of said given processing task, said predetermined task time-interval being associated with said first scheduling.

32. Apparatus as claimed in claim 28, wherein said plurality of task work values corresponding to previous executions of said given processing task are combined to calculate an exponentially decaying average work done value for said given processing task.

33. Apparatus as claimed in claim 28, comprising detecting logic operable to detecting an idle time duration value within said predetermined task time-interval and calculating a task execution deadline for said given processing task in dependence upon said task work value and said idle time duration.

34. Apparatus as claimed in claim 33, wherein said task execution deadline is calculated for each of a plurality of previous executions of said given processing task and the plurality of task execution deadlines are combined to calculate an exponentially decaying average task execution deadline value.

35. Apparatus as claimed in claim 33, wherein said target processor performance level for said given processing task is calculated in dependence upon said exponentially decaying average work done value and said exponentially decaying average task execution deadline value corresponding to said given processing task.

36. Apparatus as claimed in claim 28, further comprising:
suspended execution period detecting logic operable to detect at least one suspended execution period during processing of said given processing task, said at least one suspended execution period representing a time period during which

processing is switched from said given processing task to a further, different processing task prior to completion of said first task; and

wherein said task work value calculating logic is operable to calculate said task work value for said given processing task such that it includes processor
5 utilisation during said at least one suspended execution period.

37. Apparatus as claimed in claim 36, wherein an upper threshold for said predetermined task time-interval is set such that if said given processing task continues to execute without detection of said suspended execution period for a
10 duration greater than or equal to said upper threshold, said target processor performance level for said task is automatically recalculated.

38. Apparatus according to claim 28, wherein a flag value is stored for each task, said flag value indicating if the corresponding task has started to execute but has not
15 yet completed execution.

39. Apparatus as claimed in claim 28, wherein each task work value for a respective previous execution of said given processing task is normalised by a corresponding predetermined task time-interval when combining said task work
20 values to calculate said target processor performance level for said future execution of said task.